Eine natürliche Membranhierachie

Henning Fernau[1], Rudolf Freund[2], Markus L. Schmid[1], K.G. Subramanian[3]& Petra Wiederhold[4]

[1] Fachbereich 4 – Abteilung Informatikwissenschaften, Universität Trier D-54296 Trier, Germany
[2] Technische Universität Wien, Institut für Computersprachen, A-1040 Wien, Austria
[3] School of Computer Sciences, Universiti Sains Malaysia, 11800 Penang, Malaysia
[4] Department of Automatic Control,
Centro de Investigación y de Estudios Avanzados (CINVESTAV-IPN),
Av. I.P.N. 2508, Col. San Pedro Zacatenco, México 07000 D.F., México

Zusammenfassung Wir zeigen, dass kontextuelle Array-Grammatiken einen natürlichen Mechanismus liefern, über den eine beweisbar echte, durch die Anzahl von Membranen gegebene Hierarchie von Sprachfamilien angegeben werden kann.

1 Einführung

In der Theorie der P Systeme galt es längere Zeit als offene Frage, ein nicht-universelles Modell von P Systemen anzugeben, welches eine unendliche Hierarchie von Sprachfamilien anzugeben gestattet, die jeweils über eine Schranke auf die Anzahl der Membranen definiert ist [2, 3]. Geklärt wurde dies prinzipiell von Ibarra [1] durch Definition einer P System Variante, die Bezüge zu Zählerautomaten aufweist, wodurch sich die genannten Resultate erklären.

Kontextuelle Array-Regeln über dem Alphabet V haben die Gestalt $p = (\alpha, \beta)$, wobei α und β Abbildungen sind der Art $\mathbb{Z}^2 \to V$ mit endlichen aber disjunkten Definitionsbereichen U_α (dem sogenannten Selektor) bzw. U_β (dem Kontext). Für Arrays $\mathcal{C}_1, \mathcal{C}_2 \in V^{+2}$ gilt (intuitiv) $\mathcal{C}_1 \Longrightarrow_p \mathcal{C}_2$, falls wir in $\mathcal{C}_1$ und $\mathcal{C}_2$ ein Teilarray finden können, das dem Selektor (U_α, α) entspricht und falls die dem Kontext (U_β, β) entsprechenden Positionen nur mit dem Lückensymbol $\#$ in $\mathcal{C}_1$ gefüllt sind, sodass wir hier den Kontext (U_β, β) einfügen können, was somit $\mathcal{C}_2$ beschreibt. Wir betrachten im Wesentlichen den t-Modus bei der Ableitung; Regeln werden also solange ausgeführt, bis es nicht mehr geht.

2 P Systeme über kontextuelle Arrays

P Systeme über kontextuelle Arrays sind wie üblich aufgebaut, enthalten aber kontextuelle Array-Regeln in den einzelnen Membranen.

Anstelle einer formalen Definition betrachten wir die Sprache $L_{\text{star},4}$ der Sterne mit vier gleichlangen Armen über dem Alphabet$\{\mathtt{a},\mathtt{b},\mathtt{c}\}$:

$$L_{\text{star},4} := \left\{ \begin{array}{rcl} & \mathtt{c} & \\ & \mathtt{a} & \\ & \mathtt{b} & \\ \mathtt{b\,a\,b} & \mathtt{a} & \mathtt{b\,a\,b} \\ & \mathtt{b} & \\ & \mathtt{a} & \\ & \mathtt{b} & \end{array}, \begin{array}{rcl} & \mathtt{c} & \\ & \mathtt{a} & \\ & \mathtt{b} & \\ & \mathtt{a} & \\ & \mathtt{b} & \\ \mathtt{b\,a\,b\,a\,b} & \mathtt{a} & \mathtt{b\,a\,b\,a\,b} \\ & \mathtt{b} & \\ & \mathtt{a} & \\ & \mathtt{b} & \\ & \mathtt{a} & \\ & \mathtt{b} & \end{array}, \begin{array}{rcl} & \mathtt{c} & \\ & \mathtt{a} & \\ & \mathtt{b} & \\ & \mathtt{a} & \\ & \mathtt{b} & \\ & \mathtt{a} & \\ & \mathtt{b} & \\ \mathtt{b\,a\,b\,a\,b\,a\,b} & \mathtt{a} & \mathtt{b\,a\,b\,a\,b\,a\,b} \\ & \mathtt{b} & \\ & \mathtt{a} & \\ & \mathtt{b} & \\ & \mathtt{a} & \\ & \mathtt{b} & \\ & \mathtt{a} & \\ & \mathtt{b} & \end{array}, \ldots \right\}$$

Wir zeigen, wie ein kontextuelles P System mit linearer Membranstruktur (und fünf Membranen) diese Sprache beschreibt.

$$\Pi_{\text{star},4} = (\{\mathtt{a},\mathtt{b},\mathtt{c}\}, \#, \mu, A_1, \ldots, A_5, P_1, \ldots, P_5, 5)\,,$$

wobei $\mu = [_1\ [_2\ [_3\ [_4\ [_5\ ]_5\ ]_4\ ]_3\ ]_2\ ]_1$. Die Axiomenmengen sind gegeben durch:

$$A_1 = \left\{ \begin{array}{rcl} & \mathtt{a} & \\ & \mathtt{b} & \\ \mathtt{a\,b} & \mathtt{a} & \mathtt{b\,a} \\ & \mathtt{b} & \\ & \mathtt{a} & \end{array} \right\} \text{ and } A_2 = A_3 = A_4 = A_5 = \emptyset\,,$$

Die Regeln sind definiert als:

$$\begin{aligned}
P_1 &:= \{p_{1,1}, p_{1,2}\} := \left\{ \left(\begin{array}{c} \mathtt{b} \\ \boxed{\mathtt{a}} \\ \boxed{\mathtt{b}} \end{array}\!,\ \text{in} \right), \left(\begin{array}{c} \mathtt{c} \\ \boxed{\mathtt{a}} \\ \boxed{\mathtt{b}} \end{array}\!,\ \text{in} \right) \right\}, \\
P_2 &:= \{p_{2,1}, p_{2,2}\} := \left\{ \left(\boxed{\mathtt{b}}\,\boxed{\mathtt{a}}\,\mathtt{b},\ \text{in} \right), \left(\boxed{\mathtt{a}}\,\boxed{\mathtt{b}}\,\mathtt{a},\ \text{out} \right) \right\}, \\
P_3 &:= \{p_{3,1}, p_{3,2}\} := \left\{ \left(\begin{array}{c} \boxed{\mathtt{b}} \\ \boxed{\mathtt{a}} \\ \mathtt{b} \end{array}\!,\ \text{in} \right), \left(\begin{array}{c} \boxed{\mathtt{a}} \\ \boxed{\mathtt{b}} \\ \mathtt{a} \end{array}\!,\ \text{out} \right) \right\}, \\
P_4 &:= \{p_{4,1}, p_{4,2}\} := \left\{ \left(\mathtt{b}\,\boxed{\mathtt{a}}\,\boxed{\mathtt{b}},\ \text{in} \right), \left(\mathtt{a}\,\boxed{\mathtt{b}}\,\boxed{\mathtt{a}},\ \text{out} \right) \right\}, \\
P_5 &:= \{p_{5,1}\} := \left\{ \left(\begin{array}{c} \mathtt{a} \\ \boxed{\mathtt{b}} \\ \boxed{\mathtt{a}} \end{array}\!,\ \text{out} \right) \right\}.
\end{aligned}$$

Die Idee ist, dass die Arrays während einer Berechnung ständig zwischen den den Membranen 1 und 5 hin- und herpendeln. $\mathtt{c}$ dient lediglich zum Anhalten dieser Berechnung.

3 Das Hierarchie-Resultat

In ähnlicher Weise ist auch die Kammsprache $L_{\mathrm{comb},k}$ aufgebaut, welche k "Zähne" enthält in folgender Weise:

$$\begin{array}{ccccccccc}
 & & & & & & & & \mathtt{X} \\
 & & & & & & & & \mathtt{b}_3 \\
 & & & & & & & & \mathtt{b}_2 \\
 & & & & & & & & \mathtt{b}_1 \\
 & & & & & & & & \vdots \\
\mathtt{a}_1 & & \mathtt{a}_2 & & \mathtt{a}_3 & \dots & \mathtt{a}_k & & \vdots \\
\vdots & & \vdots & & \vdots & \vdots & \vdots & & \mathtt{b}4 \\
 & & & & & & & & \mathtt{b}3 \\
\mathtt{a}_1 & & \mathtt{a}_2 & & \mathtt{a}_3 & \dots & \mathtt{a}_k & & \mathtt{b}_2 \\
\mathtt{a}_1 & \mathtt{X} & \mathtt{a}_2 & \mathtt{X} & \mathtt{a}_3 & \dots & \mathtt{a}_k & \mathtt{X} & \mathtt{b}_1
\end{array}.$$

Konkreter gilt: $L_{\mathrm{comb},4}$ ist gleich

$$\left\{
\begin{array}{ccccccccc}
 & & & & & & & & \mathtt{X} \\
 & & & & & & & & \mathtt{b}_3 \\
\mathtt{a}_1 & & \mathtt{a}_2 & & \mathtt{a}_3 & & \mathtt{a}_4 & & \mathtt{b}_2 \\
\mathtt{a}_1 & \mathtt{X} & \mathtt{a}_2 & \mathtt{X} & \mathtt{a}_3 & \mathtt{X} & \mathtt{a}_4 & \mathtt{X} & \mathtt{b}_1
\end{array},
\begin{array}{ccccccccc}
 & & & & & & & & \mathtt{X} \\
 & & & & & & & & \mathtt{b}_3 \\
 & & & & & & & & \mathtt{b}_2 \\
 & & & & & & & & \mathtt{b}_1 \\
 & & & & & & & & \mathtt{b}_4 \\
\mathtt{a}_1 & & \mathtt{a}_2 & & \mathtt{a}_3 & & \mathtt{a}_4 & & \mathtt{b}_3 \\
\mathtt{a}_1 & & \mathtt{a}_2 & & \mathtt{a}_3 & & \mathtt{a}_4 & & \mathtt{b}_2 \\
\mathtt{a}_1 & \mathtt{X} & \mathtt{a}_2 & \mathtt{X} & \mathtt{a}_3 & \mathtt{X} & \mathtt{a}_4 & \mathtt{X} & \mathtt{b}_1
\end{array},
\begin{array}{ccccccccc}
 & & & & & & & & \mathtt{X} \\
 & & & & & & & & \mathtt{b}_3 \\
 & & & & & & & & \mathtt{b}_2 \\
 & & & & & & & & \mathtt{b}_1 \\
 & & & & & & & & \mathtt{b}_4 \\
 & & & & & & & & \mathtt{b}_3 \\
 & & & & & & & & \mathtt{b}_2 \\
 & & & & & & & & \mathtt{b}_1 \\
\mathtt{a}_1 & & \mathtt{a}_2 & & \mathtt{a}_3 & & \mathtt{a}_4 & & \mathtt{b}_4 \\
\mathtt{a}_1 & & \mathtt{a}_2 & & \mathtt{a}_3 & & \mathtt{a}_4 & & \mathtt{b}_3 \\
\mathtt{a}_1 & & \mathtt{a}_2 & & \mathtt{a}_3 & & \mathtt{a}_4 & & \mathtt{b}_2 \\
\mathtt{a}_1 & \mathtt{X} & \mathtt{a}_2 & \mathtt{X} & \mathtt{a}_3 & \mathtt{X} & \mathtt{a}_4 & \mathtt{X} & \mathtt{b}_1
\end{array}, \ldots .
\right\}$$

Wir konnten zeigen:

- Es gibt ein kontextuelles Array P System mit $k+1$ Membranen, welches $L_{\mathrm{comb},k}$ beschreibt.
- $L_{\mathrm{comb},k}$ lässt sich jedoch nicht durch ein kontextuelles Array P System mit nur $\lfloor \frac{k}{3} \rfloor$ vielen Membranen erzeugen.

Dies zeigt die behauptete unendliche Hierarchie von Sprachfamilien. Der Beweis von Punkt zwei arbeitet mit dem Schubfachprinzip, aufbauend auf einer Kategorisierung der möglichen Regeln.

Eine interessante offene Frage ist, ob die unendliche Hierarchie auch von Stufe zu Stufe echt ist. Dies konnten wir an einzelnen Beispielen nur für kleine Anzahlen von Membranen beweisen.

Wir möchten abschließend noch darauf hinweisen, dass die Ergebnisse anders ausfallen, wenn Selektoren das Lückensymbol enthalten dürfen. Dann kann die Sprache $L_{\mathrm{comb},k}$ bereits mit zwei Membranen erzeugt werden. Ebenso ist noch offen, inwieweit die Alphabetgröße auf eine Konstante beschränkt werden kann.

Wir danken der Förderung des vierten Autors durch das Projekt "Universität der Großregion", welches einen Forschungsaufenthalt an der Universität Trier ermöglichte.

Literatur

1. O. H. Ibarra. The number of membranes matters. In C. Martín-Vide, G. Mauri, Gh. Paun, G. Rozenberg, and A. Salomaa, editors, *Workshop on Membrane Computing, WMC 2003*, volume 2933 of *LNCS*, pages 218–231. Springer, 2004.
2. C. Martín-Vide and Gh. Păun, editors. *Recent Topics in Mathematical and Computational Linguistics.* Editura Academiei Române, 2000.
3. The P systems Web page, `http://ppage.psystems.eu/`.